



TP de Especificación

Trabajo Práctico Grupal

30 de Marzo de 2022

Algoritmos y Estructuras de Datos I

Grupo 10

Integrante	LU	Correo electrónico
Dominguez, Emilia	37752993	maemiliadominguez@gmail.com
Kerbs, Octavio	64/22	octaviokerbs@gmail.com
Russo, Gabriel	107/19	gabrielrussoguiot@gmail.com
Traverso, Lucas	479/18	lucas6246@gmail.com



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2610 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (+54 +11) 4576-3300

<http://www.exactas.uba.ar>

Índice

1. Definición de Tipos	2
2. Problemas	3
2.1. Parte I: Juego básico	3
2.1.1. Ejercicio 1	3
2.1.2. Ejercicio 2	3
2.1.3. Ejercicio 3	3
2.1.4. Ejercicio 4	3
2.1.5. Ejercicio 5	3
2.1.6. Ejercicio 6	4
2.2. Parte II: Despejar los vacíos	5
2.2.1. Ejercicio 7	5
2.2.2. Ejercicio 8	5
2.3. Parte III: Jugador automático	6
2.3.1. Ejercicio 9	6

1. Definición de Tipos

type $pos = \mathbb{Z} \times \mathbb{Z}$

type $tablero = seq\langle seq\langle Bool \rangle \rangle$

type $jugadas = seq\langle pos \times \mathbb{Z} \rangle$

type $banderitas = seq\langle pos \rangle$

2. Problemas

2.1. Parte I: Juego básico

2.1.1. Ejercicio 1

$\text{aux minasAdyacentes } (t: \text{tablero}, p: \text{pos}) : \mathbb{Z} = \sum_{i=\max(p_0-1,0)}^{\min(p_0+1,|t|-1)} \sum_{j=\max(p_1-1,0)}^{\min(p_1+1,|t|-1)} (\text{if } (t[i][j] = \text{true}) \text{ then } 1 \text{ else } 0 \text{ fi}) -$
 $(\text{if } (t[p_0][p_1] = \text{true}) \text{ then } 1 \text{ else } 0 \text{ fi});$
 $\text{aux max } (x, y: \mathbb{Z}) : \mathbb{Z} = \text{if } (x < y) \text{ then } y \text{ else } x;$
 $\text{aux min } (x, y: \mathbb{Z}) : \mathbb{Z} = \text{if } (x < y) \text{ then } x \text{ else } y;$

2.1.2. Ejercicio 2

```
pred juegoValido (t: tablero, j: jugadas) {  
    tableroValido(t) ∧  
    posicionValida(t, j) ∧  
    jugadasNoRepetidas ∧  
    ((∀ i : ℤ)(0 ≤ i < |j| →L ((j[i]1 = minasAdyacentes(t, j[i]0)) ∧ maximaCantidadBombasEnJugada(t, j, 1)))  
}  
  
pred tableroValido (t: tablero) {  
    (∀ i : ℤ)(0 ≤ i < |t| = t[i])  
}  
  
pred posicionValida (t: tablero, j: jugada) {  
    (∀ i : ℤ)(0 ≤ i < |j| →L (0 ≤ j[i]00 < |t| ∧ 0 ≤ j[i]01 < |t|))  
}  
  
pred jugadasNoRepetidas (j: jugadas) {  
    (∀ i : ℤ)(∀ k : ℤ)((0 ≤ i < |j| ∧ 0 ≤ k < |j| ∧ i ≠ k) →L j[i]0 ≠ j[k]0)  
}  
  
pred maximaCantidadBombasEnJugada (t: tablero, j: jugadas, x: ℤ) {  
    ∑i=0|j|-1  
}
```

2.1.3. Ejercicio 3

```
proc plantarBanderita (in n: ℤ, in j: jugadas, in p: pos, inout b: banderitas) = res :  
    {  
    }
```

2.1.4. Ejercicio 4

```
proc perdió (in t: tablero, in j: jugadas, out res: Bool) {  
    }
```

2.1.5. Ejercicio 5

```
proc ganó (in t: tablero, in j: jugadas, out res: Bool) {  
    }
```

2.1.6. Ejercicio 6

```
proc jugar (in t: tablero , in b: banderitas, in p: pos, inout j: jugadas) {  
}
```

2.2. Parte II: Despejar los vacíos

2.2.1. Ejercicio 7

```
pred caminoLibre (t: tablero, p0: pos, p1: pos) {  
  
}
```

2.2.2. Ejercicio 8

```
proc jugarPlus (in t: tablero , in b: banderitas, in p: pos, inout j: jugadas) {  
}
```

2.3. Parte III: Jugador automático

2.3.1. Ejercicio 9

```
proc sugerirAutomático121 (in n:  $\mathbb{Z}$ , in b: banderitas, out p: pos) {  
  }
```